SIMILARITY SEARCHING FOR DOCUMENTS

Field of Invention

The present invention relates to a method and means for searching to find similar documents in response to a query. The invention is particularly relevant to the use of one document as a query for a search to obtain similar documents.

Description of Prior Art

Similarity searching in databases of electronically stored documents is an important area of practical application. Such searching is well known for text. Typically, the input for such searching would be a text string, and the engine would then search the database matching entries against the text string and return entries with an acceptable similarity threshold. Similar searching is available for images - an example is the IBM Corporation QBIC (Query by Image Content) package, described at and available from http://wwwqbic.almaden.ibm.com/.

Research has also been done on using structural analysis of a document in searching, particularly at the German Research Center for Artificial Intelligence GmbH (DFKI) in systems such as Office Maid and SALT. These systems are further described at http://www.dfki.uni-kl.de.

Existing techniques are effective when the query is of essentially one data type: a text string only, or an image only. In general, however, an electronic document will consist of a combination a number of data types: a typical document might contain one or more text passages, one or more images, and line art. The text passages may also be readily sub-dividable into different types, such as headings, legends, and bulk text. Using existing techniques as indicated above, similarity searching will involve extraction of one element in a particular data type followed by similarity searching appropriate to that data type.

An example of such a sequential approach is found in US Patent No. 6,002,798. This provides for an initial structural analysis of a document into areas of different type: not simply into image plus text, but also into areas of different functional significance (eg title, heading, text block). This structural information is then used to allow user searching and text indexing in chosen functional elements of the document. This mechanism is particularly useful for making the problem of text searching in complex documents more tractable - it is not, however, effective to allow searching for documents which are as a whole similar to a query document.

It is desirable to provide methods of similarity searching which allow the features of the document to be used appropriately in a search that is properly representative of the full document.

Summary of Invention

Accordingly, in a first aspect the invention provides method of searching a database to find documents similar to a query document, comprising: decomposing the query document into elements of different data types; for one or more of the elements in a first data type, conducting a first data type similarity search to return match results from the database for the one or more elements in the first data type; for one or more of the elements in a second data type, conducting a second data type similarity search to return match results from the database for the one or more elements in the second data type; combining the match results from the first data type similarity search and the second data type similarity search to provide query document match results.

Advantageously, results from each query document match may be combined to allow progressive refinement of queries using any of the data types either singly or in further combination.

In a second aspect, the invention provides a method of searching a database to find documents similar to a query document, comprising: decomposing the query document into elements of different data types; determining a layout element in a layout datatype from the spatial arrangement of the elements in the document; for the

layout element, conducting a layout similarity search to return match results from the database for the layout element.

Brief Description of Figures

Specific embodiments of the invention are described below, by way of example, with reference to the accompanying drawings, of which:

Figure 1 shows a typical document page containing different data types;

Figure 2 shows steps in a method according to an embodiment of a first aspect of the invention for conducting a similarity search for the document shown in Figure 1;

Figure 3 shows the representation of the document shown in Figure 1 as a layout of datatypes, and indicates a search step usable in a further embodiment of the method of the invention; and

Figure 4 shows steps in a method according to an embodiment of the second aspect of the invention for conducting a similarity search for layout information.

Description of Embodiments

A typical document contains a plurality of data types. The most basic data types are text and images. Document 1 shown in Figure 1 contains a text block 12 - this text block is data in a first data type. Document 1 also contains two different kinds of image. One kind, image block 13, is a photographic image, typically consisting of an array of pixels in which each pixel has a colour value. The other kind, line art block 11, is also an image but a "drawn" one, readily representable as a combination of geometric or formulaic elements - and as such, typically readily scalable. Photographic images and line art images (hereafter "pictures" and "graphics") respond differently to different image processing and analysis techniques, and are most effectively treated as different data types. Moreover, pictures and graphics will generally serve a different purpose in a document, so it is also practical for the purpose of similarity searching to treat pictures and graphics separately.

The steps involved in similarity searching for the document of Figure 1 according to an embodiment of the first aspect of the invention are shown in Figure 2.

Firstly the document 1 is selected in step 21. For an electronic document, this could be achieved through any appropriate application capable of supporting the file type or file types of the document. For a physical document, this could be achieved by scanning the document using a scanner.

Secondly in step 22, the document is decomposed into separate elements: in the case of document 1, these elements are graphic block 11, text block 12, and picture block 13. In the case of text block 12, it is desirable for optical character recognition to be carried out at this point so that the text block element resulting from decomposition consists of ASCII text. Decomposition of the document is achieved by an analysis and recognition process through which the different parts of the document are recognised as being text, pictures or graphics. Decomposition of a document into separate data types in this way is known, using for example techniques identified in "Block Segmentation and Text Extraction in Mixed Text/Image Documents" by FM Wahl, KY Wong and RG Casey, Computer Graphics and Image Processing, Vol. 20 (1982) (a further example is provided in US Patent No. 6,002,798). Software adapted for use with proprietary scanners to decompose the elements of a scanned page into separate data types (in order to optimise the scanning process for each data type) is provided by Hewlett-Packard Company as "HP PrecisionScan". The output of HP PrecisionScan is a set of elements each in a single data type, each of which can be selected for further processing.

The result of decomposition is a set of elements, each element having a single data type. For a particular data type, such as text, then either all text is determined to be part of a single element, or else physically distinct areas of text are considered as separate elements, depending on how the decomposition is carried out. In one version of the embodiment all the elements of the document are used in similarity searching: in other versions one or more of the elements are selected for use in similarity searching (or the user is even allowed an opportunity to select part of an element for such further processing).

Separate elements are then used in similarity searching 23, 24 against a database, for example a database representing content available on the World Wide Web. Should all the elements be of one data type, this reduces to a conventional similarity searching problem addressable with a single search engine for the relevant data type. However, if elements are of different data types, then separate search engines are used for each data type. Appropriate search engines for similarity searching for different data types are known. For example, for text, appropriate linguistic matching toolkits are available from Teragram Corporation (http://www.teragram.com) and Inxight Software, Inc. (http://www.inxight.com/). In each case an appropriate preconditioning step 23 is desirable before the matching step 24, as will be discussed briefly in relation to the main data types below.

For example, Inxight Summarizer is a software component technology that summarises a document by extracting key sentences from the document. This is the preconditioning step 23. These summaries can then be matched against each other in the matching step 24. Inxight Summarizer generates indicative summaries that contain key sentence. elements from a document.The essence of the text isextractedby stemming and text normalisation technology to obtain a concise and canonical synopsis of the text. "Stemming" is the replacement of a word by its root and part-of-speech (e.g. "I had wanted" -> "to want/first person/pluperfect"), whereas "normalisation" involves replacement of one of several forms with a "concept" (e.g. "2/3/99, Feb 2nd,1999 and 2nd February" are all alternate forms of the same concept). The matching step 24 can then be carried out on the stemmed and normalised results of the preconditioning step 23 with confidence that text content which is genuinely similar will be matched without adverse influence from unwanted syntax considerations.

An example of an image searching tool is the IBM QBIC package, as indicated above. QBIC is further described at http://wwwqbic.almaden.ibm.com/. This package is adapted to precondition the images by analysing for a number of different criteria, such as colour percentages, colour layout, and textures occurring in the images. These criteria are then used in combination in a matching step 24. There are many

other known applications of "searching a 'new' image for known objects, from robot vision (a robot searching for parts in a bin), through to traffic monitoring systems (automatic detection of car license plates) - the present matching problem is essentially the inverse of these known problems.

It can be appreciated also that a serial approach could be used effectively: for example, first using a "straight edge" histogram to enable differentiation between natural and artificial scenes; then using an "edge length" histogram (an shortage of long edges probably indicates a natural scene); testing for a large area of blue tone at the top of the image (indicating an outdoor scene); and testing for significant elements of flesh tones", indicating that there is an image containing representations of people - which can be followed by a face matching analysis to find the same faces. Clearly a combination of serial and parallel steps can be employed.

The result of the similarity searching is a set of series of matching scores for documents in the database, such a set existing for each element searched. Each of these search scores needs to be normalised 25 for combination 26 to achieve a combined search result 27. The normalisation step 25 is to ensure that a correct balance is given to the results of the different searching steps 24. This can either be to weight each element of the document equally, to weight each element of the document according to its perceived importance in the document, or according to a user assessment of the relative importance of the different elements of the document.

A preferred solution may involve a mixture of automatic and manual weighting. A particularly effective approach is to use synopsis generation techniques on the textual part to produce a set of textual search criteria and also to present a set of possible criteria based on the non-textual parts. These criteria are then presented to the user for verification. Such a user based approach is easy to use (and it is also easy for a user to tell when it is ineffective). For example, auser may be asked if he/she wanted to search for things that matched the textual synopses, or, for the image and drawing parts, whether he wanted "this person", "scenes like this", "pictures containing this object"... or "pages that look like this one".

The combined result 27 is as for conventional similarity searching: a series of matching scores (generally expressed as percentages) listing documents in the database from best towards worst matches.

Generally, most effective user querying will be achieved where it is possible for the user to achieve successive refinement of the user query - using the results of one round of querying as a basis for constructing the next round of querying - so in practice the combined result 27 will frequently be fed back to a later selection step to allow effective iterative searching.

Further use can be made of information derived from page decomposition in similarity searching. In addition to the separate elements provided by page decomposition (graphic 11, text block 12, and picture 13), further information is provided in the arrangement of the different elements within the document. As is shown in Figure 3, a further output available from page decomposition is a data type plan 31 representing the document as a line art block, a text block, and an image block, arranged vertically in sequence - decomposition into layouts is discussed is US Patent No. 6,002,798. However, the present inventors have appreciated that this data type plan can itself be used as a layout data type. This allows yet another element - the layout data type element - to be used in searching 32 of a database (provided that layout information is available in or derivable from the database entries). The results of similarity searching for such a layout element can be combined with similarity searches for other elements exactly as described in Figure 2., with layout data type 31 emerging from the decomposition step 22 and then being used in a searching step 32 equivalent and parallel to searching steps 23 and 24 (followed by a normalisation step before combination in step 26 with results from other data types.

In an embodiment according to the second aspect of the invention, similarity searching is conducted using the layout data type alone. The steps to be followed are essentially as in conventional similarity searching - this is shown in Figure 4, with elements common to the first aspect of the invention given the same reference numbers as in Figure 2. Layout similarity searching, whether used on its own or as one of the elements in a combined search as described in the first aspect of the invention, is more powerful if a number of different data types are used for text and

for overall document type. Using a rule-based approach, different text blocks and whole documents, especially in the case of formal workflow documents, can be assigned particular functions with relatively high confidence. For example, it is well known that isolated text blocks at the top of a page and handwriting at the bottom are suggestive of a letter, and so different spatial regions of the document can be assigned to appropriate functional fields (address, letter text etc) - likewise, table and currency totals in a document can be identified as a discrete element, and their presence limits the document to another group (bill, quote or invoice). Layout searching can thus involve matching to templates representing different workflow document types (thus promoting matching of a document determined to be a letter against other letters). An appropriate mechanism is to normalise a layout for size, orientation and skew, and then carrying out an "exclusive or" operation on the query element and the layout records in the database - this will be effective provided that all records involved have a broadly common format.

The difficulty of this problem depends on the nature and type of documents that are to be considered for matching. If the "universe" of documents is well defined, then there are tools available that can do an accurate job of classifying and labelling within that universe (e.g. OfficeMaid from DFKI). What is required in this case is classification according to a set of conventions laid down for the various classes of documents available for consideration. Conventions are here essentially rules that need not be closely followed: consequently an appropriate approach to this problem is rule based (most conveniently using fuzzy rules). Training of a neural network would also be an effective approach to adopt. The skilled person will appreciate how conventional fuzzy rule or neural network approaches could be adapted for use in a solution to this problem.

The skilled man will appreciate that modifications of the embodiments described above can readily be carried out without departing from the invention as defined in the claims.